



Towards Energy-Aware IaaS-PaaS Co-design

Alexandra Carpen-Amarie, Djawida Dib, Anne-Cécile Orgerie, Guillaume Pierre

► To cite this version:

Alexandra Carpen-Amarie, Djawida Dib, Anne-Cécile Orgerie, Guillaume Pierre. Towards Energy-Aware IaaS-PaaS Co-design. SMARTGREENS: International Conference on Smart Grids and Green IT Systems,, Apr 2014, Barcelone, Spain. hal-01088946

HAL Id: hal-01088946

<https://inria.hal.science/hal-01088946>

Submitted on 29 Nov 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Towards Energy-Aware IaaS-PaaS Co-design

Alexandra Carpen-Amarie¹, Djawida Dib¹, Anne-Cécile Orgerie² and Guillaume Pierre³

¹*INRIA - IRISA, Rennes, France*

²*CNRS - IRISA, Rennes, France*

³*University of Rennes 1 - IRISA, France*

{Alexandra.Carpen-Amarie, Djawida.Dib}@inria.fr, {Anne-Cecile.Orgerie, Guillaume.Pierre}@irisa.fr

Keywords: Cloud computing, energy-awareness, IaaS, PaaS

Abstract: The wide adoption of the cloud computing paradigm plays a crucial role in the ever-increasing demand for energy-efficient data centers. Driven by this requirement, cloud providers resort to a variety of techniques to improve energy usage at each level of the cloud computing stack. However, prior studies mostly consider resource-level energy optimizations in IaaS clouds, overlooking the workload-related information locked at higher levels, such as PaaS clouds. In this position paper, we argue that cross-layer cooperation in clouds is a key to achieving an optimized resource management, both performance and energy-wise. To this end, we claim there is a need for a cooperation API between IaaS and PaaS clouds, enabling each layer to share specific information and to trigger correlated decisions. We identify the drawbacks raised by such co-design objectives and discuss opportunities for energy usage optimizations. Moreover, we outline the design of a set of extension modules for Libcloud to serve as building blocks for cross-layer information sharing and cooperation.

1 INTRODUCTION

Energy efficiency is a major concern in large-scale cloud infrastructures. In 2010, data centers worldwide reportedly consumed 203.4 to 271.8 TWh, which accounted for 1.1% to 1.5% of the global electricity use. This number is expected to grow even further (Koomey, 2011). To help make the planet greener while reducing their energy-related costs, cloud operators use a variety of techniques such as workload consolidation and dynamic virtual machine (VM) resizing. These strategies are however fundamentally limited by the lack of knowledge by the data center infrastructure about the applications that generate the workload. If the data center infrastructure understood better the full software stack it supports, it would be able to use more aggressive consolidation policies and derive further energy savings.

Platform-as-a-Service (PaaS) environments deliver users complete runtime environments for building, deploying and hosting software applications, while dispensing them from the complexity of managing the underlying software and hardware required for their applications. PaaS systems rely on computational resources obtained on-demand from Infrastructure-as-a-Service (IaaS) clouds, by acting as

clients that can create/remove, suspend/resume, shutdown/restart, or migrate VMs.

This position paper claims that significant energy gains could be obtained by creating a cooperation API between the IaaS layer (in charge of handling elementary computing resources) and the PaaS layer (in charge of hosting applications). We discuss two complementary approaches for establishing such cooperation:

Cross-layer information sharing. Each of the two layers contains information which is potentially relevant for the other in order to reduce the energy footprint of a given application. For example, an energy-aware IaaS may have an estimate of the energy footprint of individual virtual machine types, which may be useful for the PaaS to preferentially select energy-efficient VM instance types. Conversely, a PaaS layer may build short-term traffic predictions which constitute useful hints for IaaS-level consolidation algorithms.

Cross-layer coordination. Both layers should coordinate their reconfiguration actions to reach a state where they help each other in achieving their goals rather than potentially take mutually detrimental decisions. For example, if the IaaS layer decides to migrate a specific VM, then the PaaS could in many cases temporarily reduce the load of the concerned

VM to facilitate its migration. Conversely, if the PaaS layer gives early warnings to the IaaS about future requests for creating/destroying resources, the IaaS layer can prepare in advance for these changes.

This paper proposes a research agenda towards the co-design of IaaS and PaaS layers. Section 2 first discusses the state of the art and motivation for this work. Section 3 details the issues created by the lack of IaaS-PaaS co-design. Then, Section 4 presents our early ideas about opportunities for improvement, while Section 5 outlines the design of interfaces to enable cross-layer cooperation in clouds. Finally, Section 6 concludes this work.

2 STATE OF THE ART

Many recent research works have been proposed to take into account the consumed energy in data centers. These works can be classified into two categories: (1) energy-aware systems and (2) energy efficient systems. The first category aims at informing cloud providers and users about the consumed energy for running their applications. For instance in (Singh et al., 2013), the authors propose a cloud-based customer-centric architecture that allows consumers to own and control access to their energy usage data and have it analyzed using algorithms of their choice. The second category aims at optimizing the consumed energy for running applications. Different techniques are used to achieve energy efficiency. In (Phan et al., 2012), the authors propose to increase the use of renewable energy by dynamically moving services across data centers. The works in (Jaiantilal et al., 2010; Takouna et al., 2011) propose to adjust CPU voltage and frequency according to the load for saving energy. In (Gadafi et al., 2010), the authors dynamically resize the active server set according to varying workload conditions. Workload consolidation on a limited number of servers is addressed in (Duy et al., 2010; Meisner et al., 2009), in order to allow idle servers to be switched off and save energy.

Other works have focused on analyzing the impact of various optimization techniques on energy consumption. For instance, in (Orgerie et al., 2010), the authors investigate the tradeoffs and limitations of existing energy models for large-scale systems.

3 ISSUES

3.1 Energy-awareness in IaaS and PaaS

PaaS frameworks provide application developers with runtime environments where applications can be easily deployed and managed in the cloud (Pierre and Stratan, 2012; Dib et al., 2013). They do not provide cloud resources directly, but rather make use of an underlying IaaS layer in charge of resource management. Consequently, PaaS does not have access to information about the underlying resources, while IaaS does not have access to information about the running applications. However, both IaaS and PaaS systems target multi-objective optimizations including primarily cost, performance and energy consumption. These parameters are intrinsically related, thus requiring complex trade-offs to be made between them.

Traditionally, energy-awareness and energy-efficiency have been addressed at the IaaS level with techniques presented in Section 2. To our knowledge, no research work has been targeting energy-efficiency or energy-awareness at the PaaS level. However, studies on real cloud infrastructures show that servers are considerably underutilized (Zhang et al., 2011), thus wasting large amounts of energy (Acc, 2010). This information about resource utilization is not available at the PaaS level, and nor is the energy consumption of resources. So, the PaaS layer has no means to provide energy-related metrics to users (e.g. energy-efficiency of their VMs, energy budget of their applications), and thus no way to increase their energy-awareness.

Conversely, the IaaS layer has no information about the applications and the users. As previously discussed, such knowledge would be useful to predict future resource utilization, thus leading to better energy-related decisions at the IaaS level.

3.2 The case for IaaS-PaaS co-design

We believe that the separation of the cloud stack in two distinct IaaS and PaaS layers, while having great advantages for portability and separation of concerns, is detrimental in terms of energy awareness. If each layer is allowed to take energy-related decisions independently, these uncoordinated actions can lead to significant resource waste and performance degradation, possibly negating the benefits of energy awareness altogether.

For instance, the IaaS layer can decide to migrate a virtual machine (VM) in order to perform a better consolidation for energy-efficiency purposes. Yet, this same VM may end a few seconds later because it

gets released by the PaaS layer. The decision to shut-down this VM may have been taken several minutes in advance by the PaaS layer. If this information is not communicated to the IaaS layer, we take the risk that IaaS will invest previous resources (for example by migrating the VM) without seeing any benefit from this action (because the VM gets shut down just after).

Conversely, the PaaS layer may help the IaaS layer in performing its VM management actions. For example, it is often easy at the PaaS level to temporarily redirect one VM's workload to another (by redefining load balancing parameters for example). Offloading a VM for just a few tens of seconds may greatly facilitate IaaS-level management tasks such as VM migration.

In order to avoid counterproductive independent optimizations, we believe that the IaaS and the PaaS should share their energy-related information and coordinate their reconfiguration actions. This coordination aims at allowing system-level optimizations and trade-offs.

4 OPPORTUNITIES

In order to design efficient cloud frameworks, both performance and energy-wise, it is essential to understand how the cloud stack layers can interact and cooperate. We argue that there is a need to extend the traditional boundaries of each class of cloud computing frameworks to ensure an effective management of resources, while delivering seamless application performance and SLA compliance to higher-level users. In this section we identify a set of design opportunities for both PaaS and IaaS providers to help address these goals. The research directions we investigate are twofold: first, we focus on the bidirectional information sharing opportunities between these cloud layers and the coordinated decisions they can trigger. Second, we discuss security concerns that arise from such interactions.

4.1 Workload-aware IaaS frameworks

The emergence of a wide array of open-source IaaS solutions led to a gradual adoption of open standards and interfaces. This was a key incentive for enterprises to deploy private clouds in their data centers, allowing them to take advantage of the flexibility and performance of virtualized infrastructures, without the vendor lock-in shortcomings of proprietary solutions.

Most open IaaS cloud frameworks deliver similar services, typically focusing on resource manage-

ment, VM lifecycle and on providing framework usage statistics. However, little has been done to achieve energy-efficient resource usage in such infrastructure clouds, despite the fact that lowering energy consumption is becoming a growing concern for data center management. This context provides a valuable opportunity for IaaS cloud providers to consider techniques that take into account the requirements of higher-level services, such as PaaS offerings, which traditionally build on infrastructure clouds. To this end, we believe that PaaS clouds can deliver a set of insights to the infrastructure level to guide its resource allocation and energy optimization decisions.

4.1.1 Passing workload information

VM allocation strategies typically rely on information concerning physical machine capabilities and their usage across the data center. Nevertheless, several factors may impact the performance of VMs, with critical side effects on application performance, user costs and power consumption. For instance, intensive network traffic between a few VMs may lower the available bandwidth across an entire cluster. On the other hand, PaaS virtual resource management services use the IaaS APIs to specify the number and type of VMs required upon application deployment. No information about the application class or its execution parameters is forwarded to the underlying infrastructure.

In this context, the answer to achieving better resource management at the infrastructure level is to enable IaaS schedulers to analyze and exploit a wider spectrum of parameters associated with the application behavior. Thus, as such workload properties are generally opaque to the underlying VM manager, a possible approach is to enable PaaS services to expose them to the infrastructure layer, as detailed below.

Resource usage patterns. The resource needs of VMs have a potentially heavy impact on the VM migration duration. As an example, it was shown that memory-bound VMs lead to very inefficient migration times (Liu et al., 2011). IaaS systems may avoid such operations by flagging affected VMs at deployment time. A more fine-grained approach may enable the PaaS services to trigger such flags dynamically, as the application enters a memory-intensive stage, and allow migration outside these intervals.

Execution time. Cloud services can accommodate a large set of application types. For instance, web servers represent long-running jobs with specific access patterns. Oppositely, a short-lived VM, such as a MapReduce job, may raise specific scheduling constraints. Thus, as many MapReduce jobs run for short periods of time, migrating the VMs during ap-

plication execution may degrade their performance to unacceptable levels (e.g., migrating the VM may take longer than its actual runtime).

Elasticity needs. Unlike the infrastructure manager, the PaaS services may be able to estimate the application requirements in terms of workload peaks. As elasticity is one of the most appealing cloud features, anticipating application needs and passing them on to the IaaS layer may result in more timely workload adaptation mechanisms, which cannot be achieved by the PaaS services alone.

4.1.2 Using workload information to reduce energy consumption

Such workload knowledge may allow IaaS providers not only to optimize resource allocation decisions, but also to improve the overall energy-consumption levels. This goal is typically achieved by adapting the number of used physical nodes to the workload and shutting down underutilized nodes (Orgerie et al., 2010). Taking into account additional PaaS-level aspects may lead to significant and accurate energy-saving decisions, as exemplified below.

Resource scheduling. Resource allocation in IaaS clouds is generally based on the VM type and straightforward scheduling policies. Beyond the VM type in terms of required CPUs and memory, an essential parameter is represented by the association of VMs with a specific virtual cluster, and thus, a single application. By taking into account this type of hint made available by PaaS-level services, an IaaS provider may change the allocation process of similar VMs. For example, VMs belonging to the same job may be deployed on the same physical machines to benefit from data locality and avoid unnecessary and performance-degrading network traffic.

Node shutdown policies. A widely studied mechanism to minimize energy consumption is shutting down unused nodes. However, resource managers usually lack any knowledge related to the type of workload or about the real-time evolution of resource requirements. Such information provided by higher-level services such as PaaS providers may allow for more aggressive shutdown policies, without incurring performance degradations. When the infrastructure manager is aware of the short-term evolution of its workload, it can adopt less conservative policies to reduce the powered-on resources and the total energy usage.

Management of workload peaks. Cloud services have developed on the premise that they can provide on-demand resource adaptation to workload bursts, relieving the client from the burden of overprovisioning resources to cope with unexpected work-

load fluctuations. Without any means to foresee the resource requirements of its scheduled jobs, an IaaS cloud is constrained to maintain a pool of unused running physical machines to accommodate possible workload peaks. To minimize the impact of such an approach on energy consumption, PaaS services may anticipate workload trends and, furthermore, forward them to the infrastructure level. Thus, IaaS schedulers can redirect small bursts to public clouds when the cost of switching nodes on is more important than the cost of obtaining remote VMs, both performance and energy-wise. When facing longer-duration workload peaks, the IaaS cloud can automatically adapt the pool of available physical machines, by enabling nodes in advance.

4.2 Optimizing PaaS resource management

Whereas advice from application-aware PaaS services may improve the efficiency of resource management decisions, PaaS frameworks can also benefit from infrastructure-level insights.

4.2.1 Exposing low-level knowledge to PaaS clouds

Infrastructure details. PaaS management services interact with the underlying infrastructure through the interfaces exposed by the IaaS provider, which typically hide low-level details from the user. As this information is provided in private clouds for administrative purposes, it can be also made available for higher-level entities, such as PaaS frameworks running on the infrastructure. For instance, knowledge about cluster utilization and node properties may allow the PaaS services to select the most suitable VM types to execute an application, enabling the use of large VMs when the infrastructure comprises high-performance physical machines.

Energy awareness. The first step towards energy-efficient services at the platform level is user awareness, in particular if lowering the energy consumption is associated with monetary incentives. The user may be presented with several execution plans and the corresponding energy consumption estimations for the same application. He can thus choose the best execution schedule in terms of energy, regardless of a performance loss or a longer execution time. Nevertheless, the PaaS layer has no means to accurately estimate the energy consumption of a given application. It is at the IaaS level where such predictions can be made, based on the load of the infrastructure and the application requirements.

4.2.2 IaaS-PaaS coordinated decisions

Since PaaS services do not have full control over the management of their virtual resources, the key to an optimized resource usage and energy consumption lies in cross-layer coordination. Despite the fact that all VM allocation and placement decisions are taken at the infrastructure level, such a collaboration can lead to both improved scheduling mechanisms and a reduced incidence of conflicting actions.

Adaptive resource management. Equipped with detailed information about the user job, the IaaS layer can yield multiple scheduling options to handle it. According to user set thresholds, such as an execution deadline or a given energy budget, the IaaS resource manager may be extended to delay jobs according to the availability of energy-efficient or renewable energy sources (e.g., wind energy). PaaS may facilitate certain VM management actions as well. To reduce the time spent during VM migration, the PaaS can be instructed to temporarily decrease the load of the VM, which may entail performance and energy gains.

Preventing conflicting optimizations. Cloud cross-layer communication contributes to another unexpected, yet significant optimization angle for VM management, that is to prevent conflicting optimizations. As an example, the PaaS layer may detect an idle VM and redirect a slice of the job workload to it. Simultaneously, the infrastructure manager can decide to migrate the same idle VM, resulting in both an inefficient migration and a reduced performance for the PaaS application. To prevent such conflicting or counter-productive optimizations, it is essential to define a standardized interface and a set of parameters to be passed between the two layers, which can ensure portability and usability across cloud providers.

4.3 Privacy and security issues

As detailed before, cross-layer coordination can help increasing both energy awareness and energy efficiency of the entire cloud system. However, this coordination needs to be clearly defined as it can raise major concerns about the security and privacy of cloud providers and users. Indeed, it has been shown that having access to the energy consumption of a cloud server can allow people to guess with high probability what type of application (among various possible ones) is running in its virtual machines (Hlavacs et al., 2011). So, passing the information about energy consumption of servers is not possible for privacy's sake. It is therefore necessary to consider a coordination of these platforms that guarantees the privacy of the users and of the applications, while providing useful

information to both layers in order to save energy. To do so, it is required to identify sensitive information that cannot be communicated between the two layers: energy consumption of physical servers, location of virtual machines (allowing users to target specific physical machines in order to be co-located with specific virtual machines). Moreover, the IaaS provider and the PaaS provider can be different, and they may not want to share sensitive information about the utilization of their platform.

5 PROPOSAL

To facilitate the interaction between cloud layers while preserving the separation and the interoperability across the cloud stack, we argue there is a need for an abstraction layer proposing coordination APIs. For instance, let us take the example of the Apache Libcloud (lib, 2014) library. Libcloud consists of a set of Python libraries providing a unified abstraction across IaaS cloud APIs such as Amazon EC2, OpenStack or OpenNebula. It hides the differences between the various cloud providers by offering generic APIs for compute, storage, authentication, load balancers and DNS services. We believe such interfacing libraries can be the key of cross-layer interoperability in the cloud stack. The development of client libraries that expose APIs to promote collaborative decisions between IaaS clouds and higher-level frameworks may come as an incentive for cloud providers to exploit the potential benefits of inter-layer cooperation.

We aim at defining a set of APIs to complement existing Libcloud features with new mechanisms for designing IaaS-PaaS cooperation strategies. Our goal is twofold. First, we plan to enable IaaS clouds to expose extended infrastructure information to upper layers or applications. In this context, two sets of APIs are required to enable PaaS providers to optimize resource management, as detailed in Section 4.2. On the one hand, the IaaS provider can supply data about the infrastructure status, including cluster utilization and node properties, which can be accessed on-demand through the Libcloud API. On the other hand, the Libcloud suite may include a set of APIs dedicated to exposing infrastructure energy consumption at the higher levels, as the first step towards energy-aware PaaS and IaaS applications.

Our second goal is to add abstract triggers at the Libcloud level to enable PaaS managers to push workload-related information to the IaaS clouds. As the applications executed on VMs rented from IaaS clouds are opaque to the infrastructure level, such APIs are essential to make application information

available for the IaaS resource allocation and management mechanisms. Consequently, such APIs should allow PaaS managers to ask the IaaS provider to flag the activity of specific VMs or groups of VMs as CPU-, IO- or memory-intensive, to predict workload peaks or to provide hints regarding the execution time of groups of VMs that belong to the same job.

Such an API-driven approach to cross-layer co-operation achieves several benefits. First, each cloud provider may choose to implement only the required plugins according to the amount of information it intends to share. While private clouds may implement the whole set of energy-related and cooperation APIs, public providers may decide to share only non-sensitive data. Furthermore, an abstraction layer such as Libcloud fosters interoperability and portability between clouds, such that PaaS providers can make data available and collect infrastructure information according to predefined formats, while each IaaS cloud can provide its own interface implementation according to its internal structure and APIs.

6 CONCLUSION

The growing demand for “green” and energy-efficient data centers is a key driver in the design of cloud computing frameworks, which target not only performance, but also cost efficiency. In particular, designing virtualized environments for energy efficiency has led to a wide spectrum of optimization techniques, most of which are intended for IaaS cloud offerings. Whereas higher-level PaaS services hold key information related to workload properties and execution, they do not have direct control over the resource-management mechanisms that could trigger energy gains. In this work, we argued that there is a need for coordinated actions between IaaS and PaaS cloud layers. We believe that the conventional boundaries of the cloud stack layers can be extended to promote energy-awareness at higher levels, while enhancing low-level resource managers with detailed workload profiles to achieve fine-grained energy saving mechanisms.

We investigated a set of opportunities for both IaaS and PaaS providers to address these objectives and highlighted the tradeoffs of cross-layer information passing. As future work, our goal is to design the proposed API suite by extending Libcloud, in order to enable cross-layer cooperation between open-source IaaS and PaaS cloud implementations and to develop prototypes of cooperating cloud frameworks.

REFERENCES

- (2014). Apache Libcloud. <http://libcloud.apache.org/>.
- Acc (2010). Cloud Computing and Sustainability: The Environmental Benefits of Moving to the Cloud. Technical report, Accenture, Microsoft and WSP Report.
- Dib, D., Parlavantzas, N., et al. (2013). Meryn: open, SLA-driven, cloud bursting PaaS. In *Proc. of ORMACloud '13*, pages 1–8, New York, NY, USA. ACM.
- Duy, T., Sato, Y., et al. (2010). Performance evaluation of a Green scheduling algorithm for energy savings in cloud computing. In *2010 IEEE Int. Sym. on Par. Dist. Proc., Workshops (IPDPSW)*, pages 1–8.
- Gadafi, A., Tchana, A., et al. (2010). Autonomic energy management in clusters. In *Proc. of the 1st ACM Workshop on Green Comp.*, pages 16–21, NY, USA.
- Hlavacs, H., Treutner, T., et al. (2011). Energy consumption side-channel attack at virtual machines in a cloud. In *IEEE Int. Conf. on Cloud and Green Comp.*, pages 605–612.
- Jaientilal, A., Jiang, Y., et al. (2010). Modeling cpu energy consumption for energy efficient scheduling. In *Proc. of the 1st Workshop on Green Computing, GCM '10*, pages 10–15, New York, NY, USA. ACM.
- Koomey, J. (2011). Growth in data center electricity use 2005 to 2010. *Oakland, CA: Analytics Press*, 1.
- Liu, H., Xu, C.-Z., et al. (2011). Performance and energy modeling for live migration of virtual machines. In *Proc. of the 20th ACM Int. Sym. on High Performance Dist. Comp. (HPDC '11)*, pages 171–182, USA.
- Meisner, D., Gold, B. T., and Wenisich, T. F. (2009). PowerNap: eliminating server idle power. *SIGPLAN Not.*, 44(3):205–216.
- Orgerie, A.-C., Lefevre, L., and Gelas, J.-P. (2010). Demystifying energy consumption in grids and clouds. In *Proc. of the Int. Green Comp. Conf.*, pages 335–342.
- Phan, D. H., Suzuki, J., et al. (2012). Evolutionary multi-objective optimization for green clouds. In *Proc. of the 14th ACM Int. Conf. on Genetic and Evolutionary Computation Conf. Companion, GECCO Companion '12*, pages 19–26, New York, NY, USA.
- Pierre, G. and Stratan, C. (2012). ConPaaS: a platform for hosting elastic cloud applications. *IEEE Internet Computing*, 16(5):88–92.
- Singh, R. P., Keshav, S., et al. (2013). A cloud-based consumer-centric architecture for energy data analytics. In *Proc. of the 4th ACM Int. Conf. on Future energy systems, e-Energy '13*, pages 63–74, USA.
- Takouna, I., Dawoud, W., et al. (2011). Dynamic configuration of virtual machine for power-proportional resource provisioning. In *Proc. of the ACM Int. Workshop on Green Comp. Middleware*, pages 1–6, USA.
- Zhang, Q., Hellerstein, J., et al. (2011). Characterizing Task Usage Shapes in Google’s Compute Clusters. In *Int. Workshop on Large Scale Distributed Systems and Middleware*.